

WHAT IS CLAIMED IS:

Sub A
1. A context controller for managing multitasking in a processor, comprising:

an event recorder that records occurrences of predetermined events; and

an event acknowledger, associated with said event recorder, that acknowledges ones of said events based on an identity of a currently-active context.

2. The context controller as recited in Claim 1 further comprising an event masker, associated with said event recorder, said event acknowledger and each context executing on said processor, that masks others of said events as a function of said each context.

3. The context controller as recited in Claim 1 wherein said event recorder is embodied in at least one flip-flop within said context controller.

4/ The context controller as recited in Claim 1 further
2 comprising:

3 a foreground task controller that activates contexts
4 corresponding to foreground tasks based on priority and in response
5 to said events; and

6 a background task controller that cyclicly activates contexts
7 corresponding to said background tasks subject to activation of
8 said contexts corresponding to said foreground tasks.

5. The context controller as recited in Claim 1 further
2 comprising a background task controller that activates contexts
3 corresponding to background tasks based on numbers of instructions
4 executed by each of said background tasks.

6. The context controller as recited in Claim 1 wherein said
2 context controller places said processor in an idle state when all
3 foreground and background tasks are inactive.

7. The context controller as recited in Claim 1 further
2 comprising a foreground task controller adapted to activate a
3 context corresponding to a particular foreground task by vectoring
4 to a software-selectable memory location.

Sub A2
B2

8. A method of managing multitasking in a processor,
comprising the steps of:

recording occurrences of predetermined events; and
acknowledging ones of said events based on an identity of a
currently-active context.

9. The method as recited in Claim 8 further comprising the
step of masking others of said events as a function of each context
executing on said processor.

10. The method as recited in Claim 8 wherein said step of
recording comprises the step of changing a state of at least one
flip-flop within said context controller.

11. The method as recited in Claim 8 further comprising the
steps of:
activating contexts corresponding to foreground tasks based on
priority and in response to said events; and
cyclicly activating contexts corresponding to said background
tasks subject to activation of said contexts corresponding to said
foreground tasks.

12. The method as recited in Claim 8 further comprising the
step of activating contexts corresponding to background tasks based
on numbers of instructions executed by each of said background
tasks.

13. The method as recited in Claim 8 further comprising the
step of placing said processor in an idle state when all foreground
and background tasks are inactive.

14. The method as recited in Claim 8 further comprising the
step of activating a context corresponding to a particular
foreground task by vectoring to a software-selectable memory
location.

Sub #37
15. A processor, comprising:
2 an instruction decoder that decodes instructions received into
3 said processor and corresponding to a plurality of tasks;
4 a plurality of register sets, corresponding to said plurality
5 of tasks, that contain operands to be manipulated;
6 an execution core, coupled to said instruction decoder and
7 said plurality of register sets, that executes instructions
8 corresponding to an active one of said plurality of tasks to
9 manipulate ones of said operands; and
10 a context controller for managing multitasking in said
11 processor, including:
12 an event recorder that records occurrences of
13 predetermined events, and
14 an event acknowledger, associated with said event
15 recorder, that acknowledges ones of said events based on an
16 identity of a currently-active context.

17. The processor as recited in Claim 15 wherein said context
2 controller further includes an event masker, associated with said
3 event recorder, said event acknowledger and each context executing
4 on said processor, that masks others of said events as a function
5 of said each context.

17. The processor as recited in Claim 15 wherein said event recorder is embodied in at least one flip-flop within said context controller.

18. The processor as recited in Claim 15 wherein said context controller further includes:

a foreground task controller that activates contexts corresponding to foreground tasks based on priority and in response to said events; and

a background task controller that cyclicly activates contexts corresponding to said background tasks subject to activation of said contexts corresponding to said foreground tasks.

19. The processor as recited in Claim 15 wherein said context controller further includes a background task controller that activates contexts corresponding to background tasks based on numbers of instructions executed by each of said background tasks.

20. The processor as recited in Claim 15 wherein said context controller places said processor in an idle state when all foreground and background tasks are inactive.

21. The processor as recited in Claim 15 wherein said context
2 controller further includes a foreground task controller adapted to
3 activate a context corresponding to a particular foreground task by
4 vectoring to a software-selectable memory location.

22. The processor as recited in Claim 15 wherein said
2 processor forms a portion of a general-purpose computer.